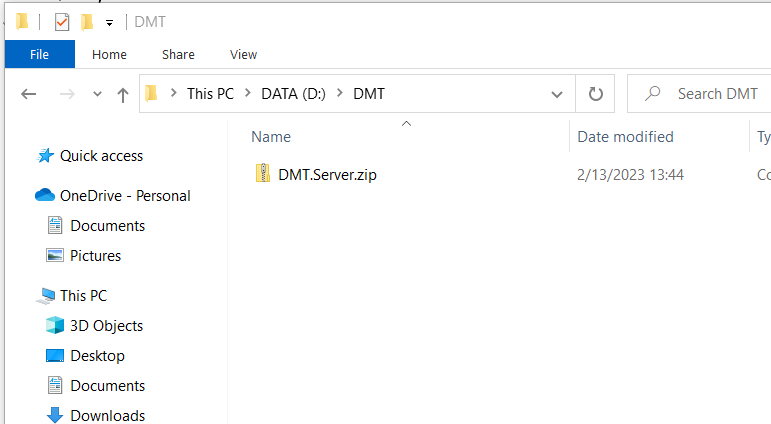
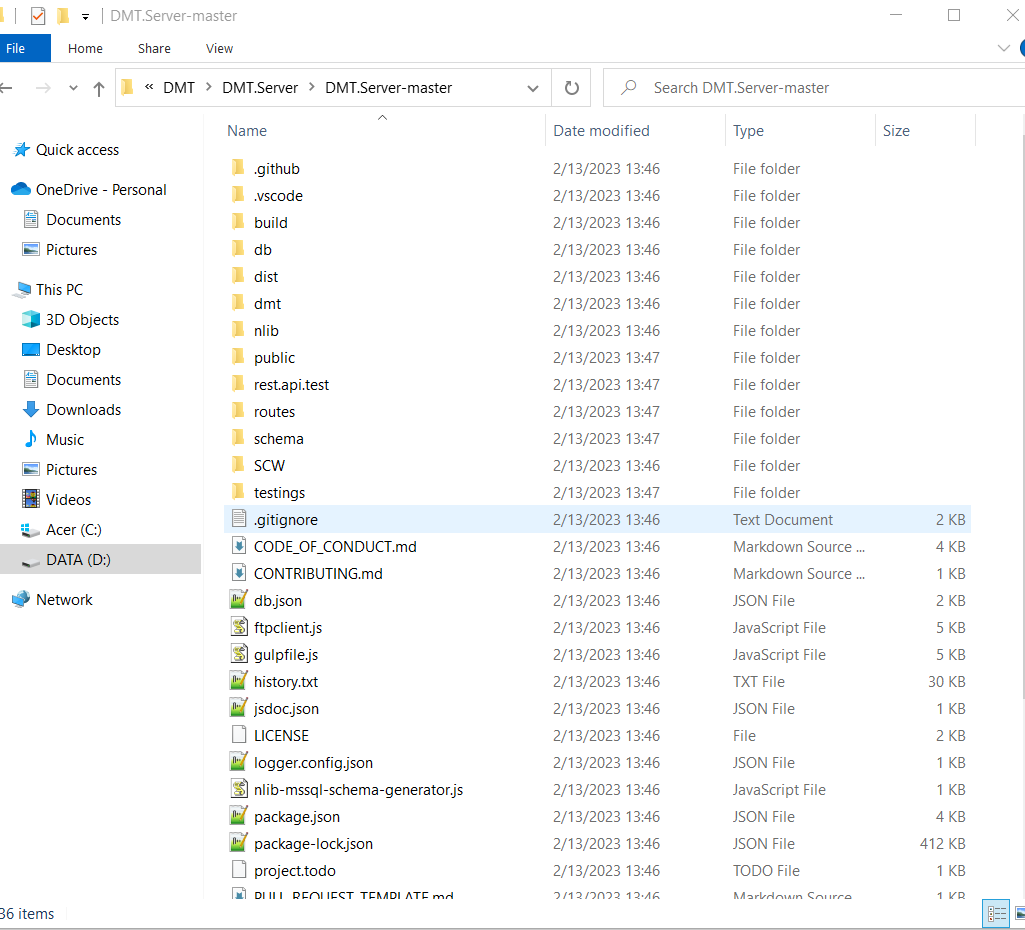
**List of source code**

Source code ของโปรแกรม TA Server จะอยู่ในรูปแบบ zip file ซึ่งเมื่อทำการแตก file แล้วจะมี source files ต่าง ๆ ซึ่งเป็น javascript files ดังรูป





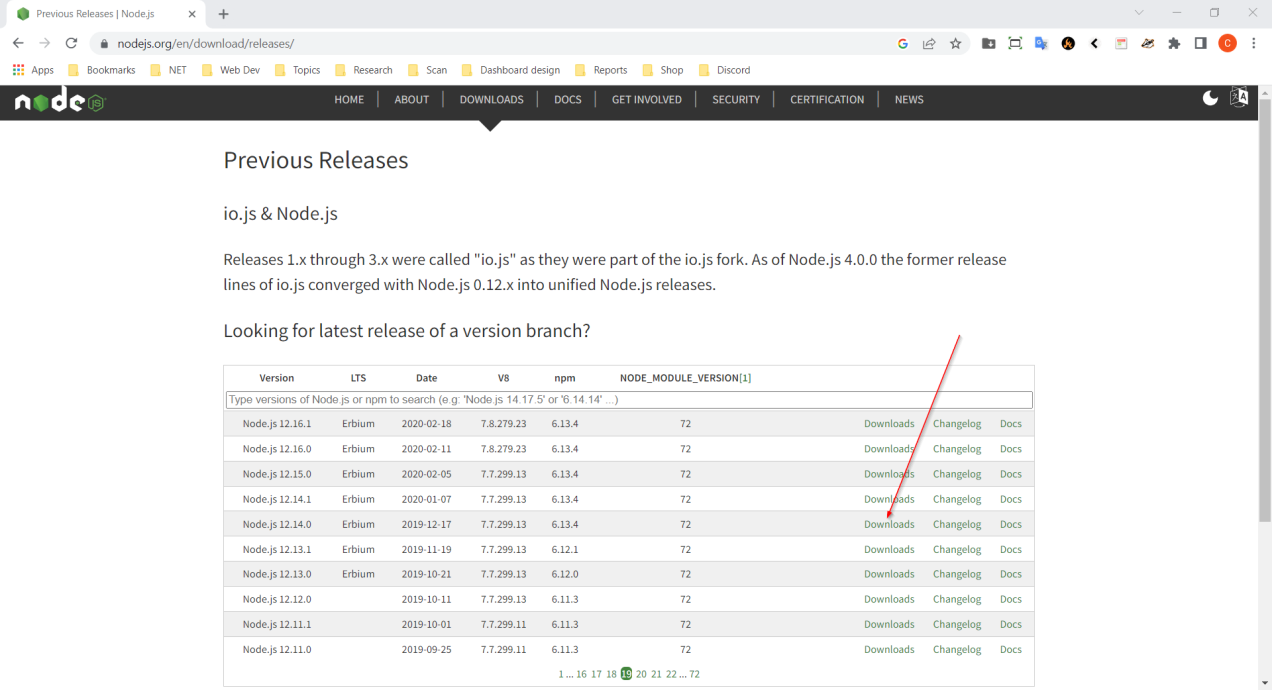
รูปแสดงรายการ folders และ files ต่าง ๆ หลังแตก zip file ออกมา

**การติดตั้ง Editor, Compiler, Component**

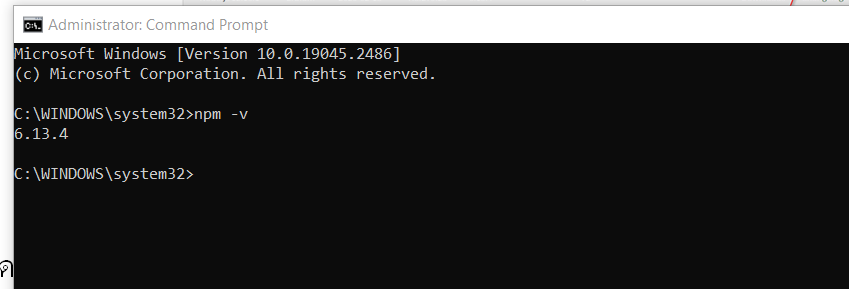
โปรแกรม TA Server เป็นโปรแกรมที่ทำหน้าที่เป็น REST API Server โดยพัฒนาด้วย node.js ซึ่งจำเป็นต้องทำการติดตั้ง node.js เสียก่อน และในส่วนของ Editor จะใช้ Microsoft Visual Studio Code ในการพัฒนา โดยการติดตั้งแบ่งเป็น

* การติดตั้ง Node.js

ทำการเปิด Web Browser และไปยัง url ดังต่อไปนี้ <https://nodejs.org/en/download/releases/> จากนั้นให้ทำการหา node.js version 12.14.0 ตามรูป



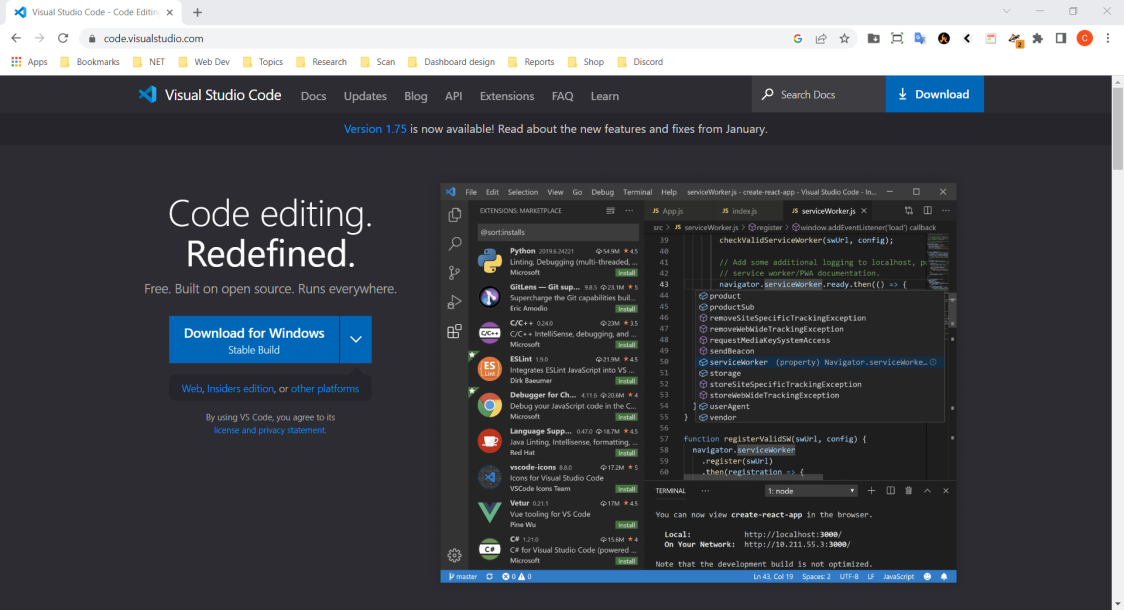
โดย เมื่อทำการ download ตัวโปรแกรม installer ที่มีชื่อว่า node-v12.14.0-x64.msi มาแล้วทำการ install บนเครื่อง โดยเมื่อทำการติดตั้งเสร็จแล้ว สามารถตรวจสอบ ว่าติดตั้งได้สมบูรณ์หรือไม่โดยการเปิด command prompt ขึ้นมา แล้วพิมพ์คำสั่งตามรูป



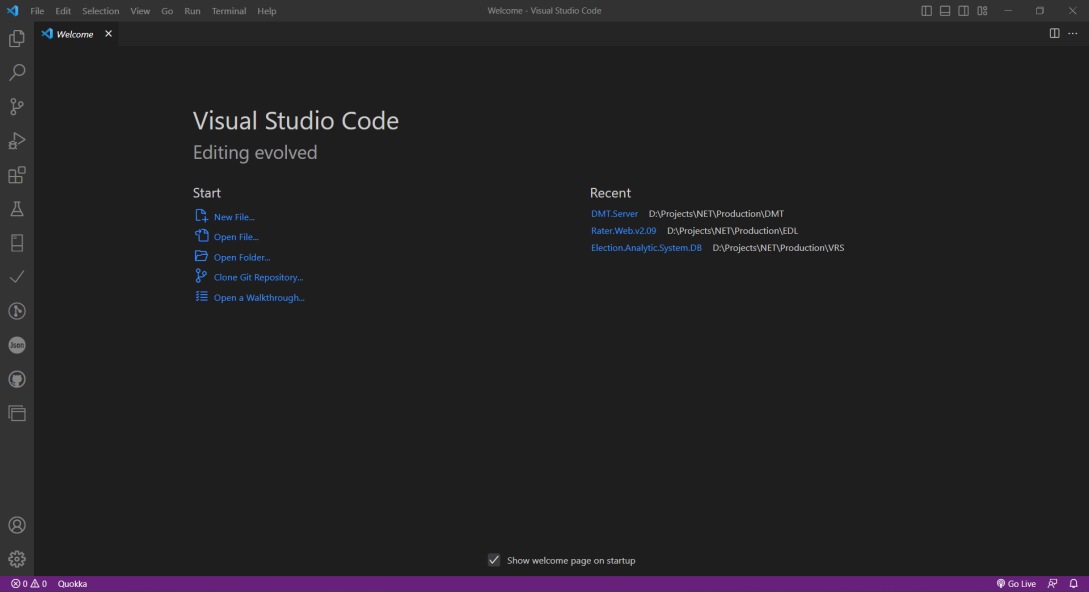
หากหน้าจอแสดงตัวเลข version ตามรูป ก็จะสามารถใช้งาน node.js ได้แล้ว

* การติดตั้ง Microsoft Visual Studio Code

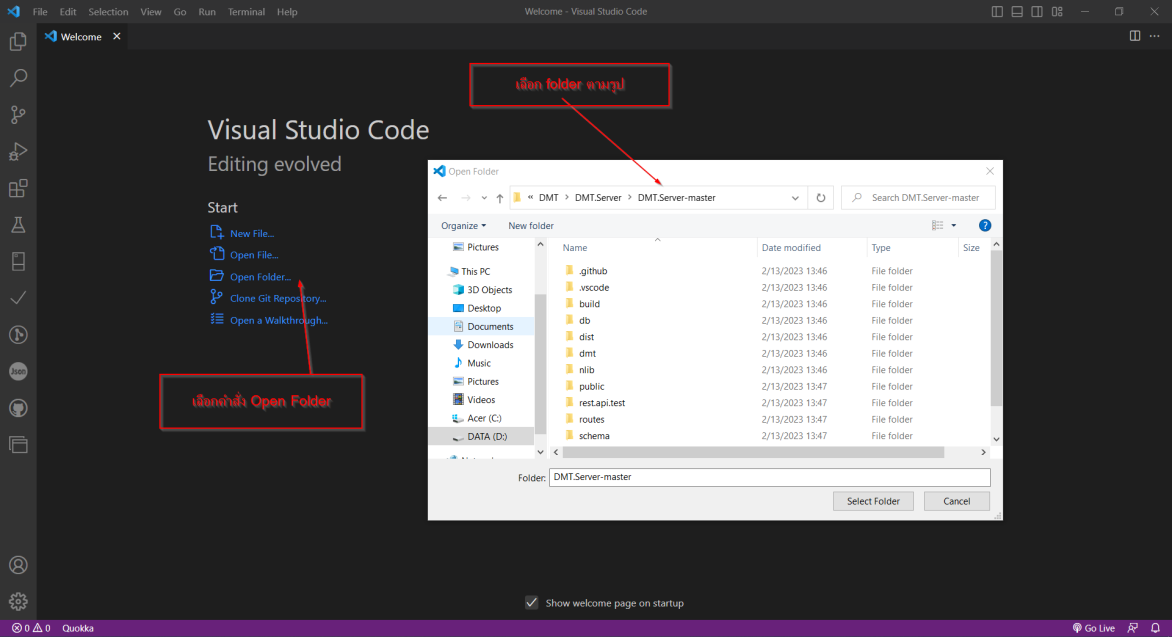
ในการติดตั้ง Visual Studio Code ให้ไปยัง Website ตาม url <https://code.visualstudio.com/> จากนั้นทำการ download โปรแกรม installer ตามรูป



เมื่อทำการติดตั้งสำเร็จ ให้เปิดโปรแกรมขึ้นมา โดยโปรแกรมจะแสดงหน้าต่างดังรูป



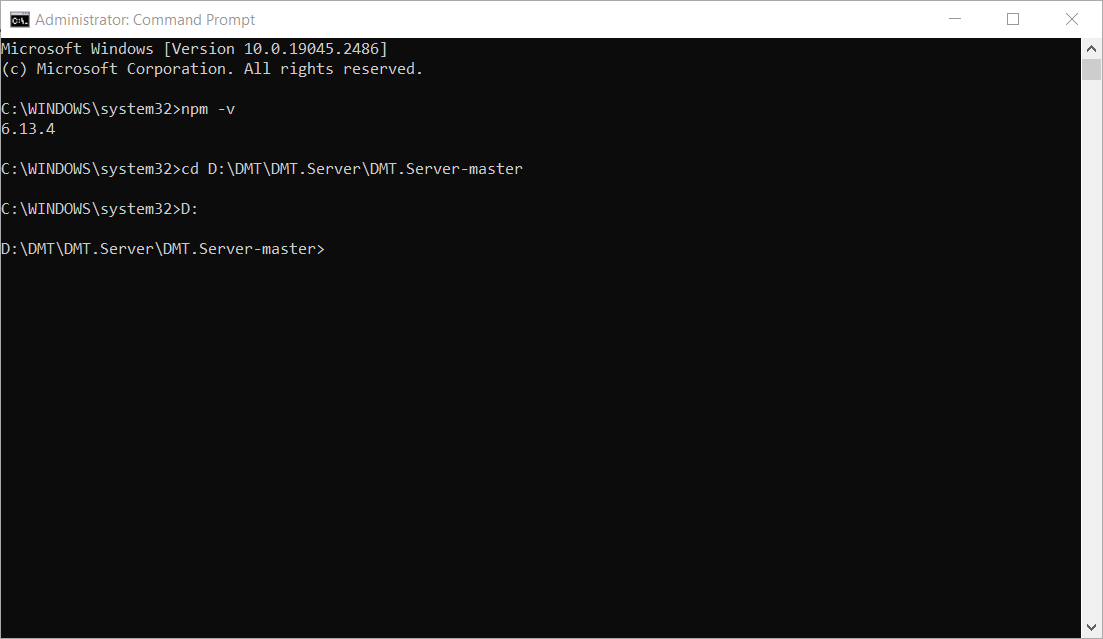
จากนั้นให้ทำการ เลือก Open Folder และทำการเลือก Folder ที่มี source code อยู่ โปรแกรมจะทำการ load source จาก folder ดังกล่าวขึ้นมา โดยจะแสดงให้เห็นดังรูป



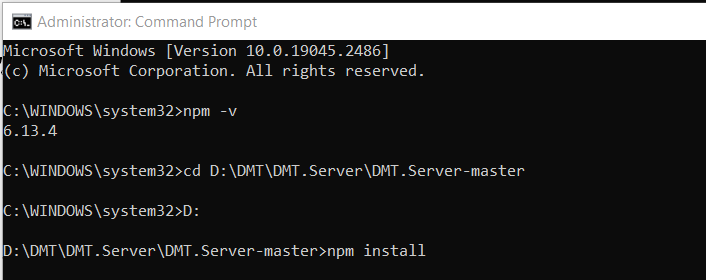
ถ้าโปรแกรมแสดงได้ตามรูป ก็จะสามารถทำงานได้

**Framework**

ภายหลังจากการติดตั้ง software ต่าง ๆ แล้ว ให้ ทำการเปิด command prompt โดยทำการเปลี่ยน directory ไปยัง folder ที่เก็บ source code อยู่ ดังรูป



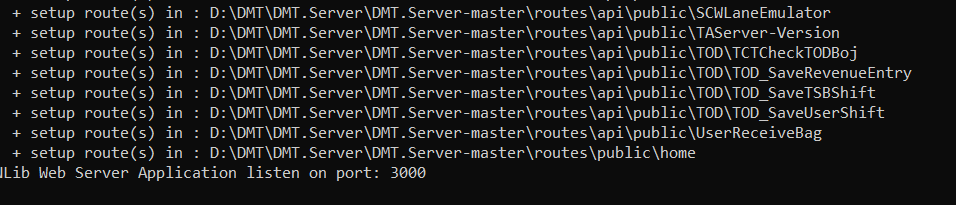
จากนั้น ให้ทำการติดตั้ง package ของ node.js ที่จำเป็นต้องใช้ โดยพิมพ์คำสั่ง npm install ดังรูป



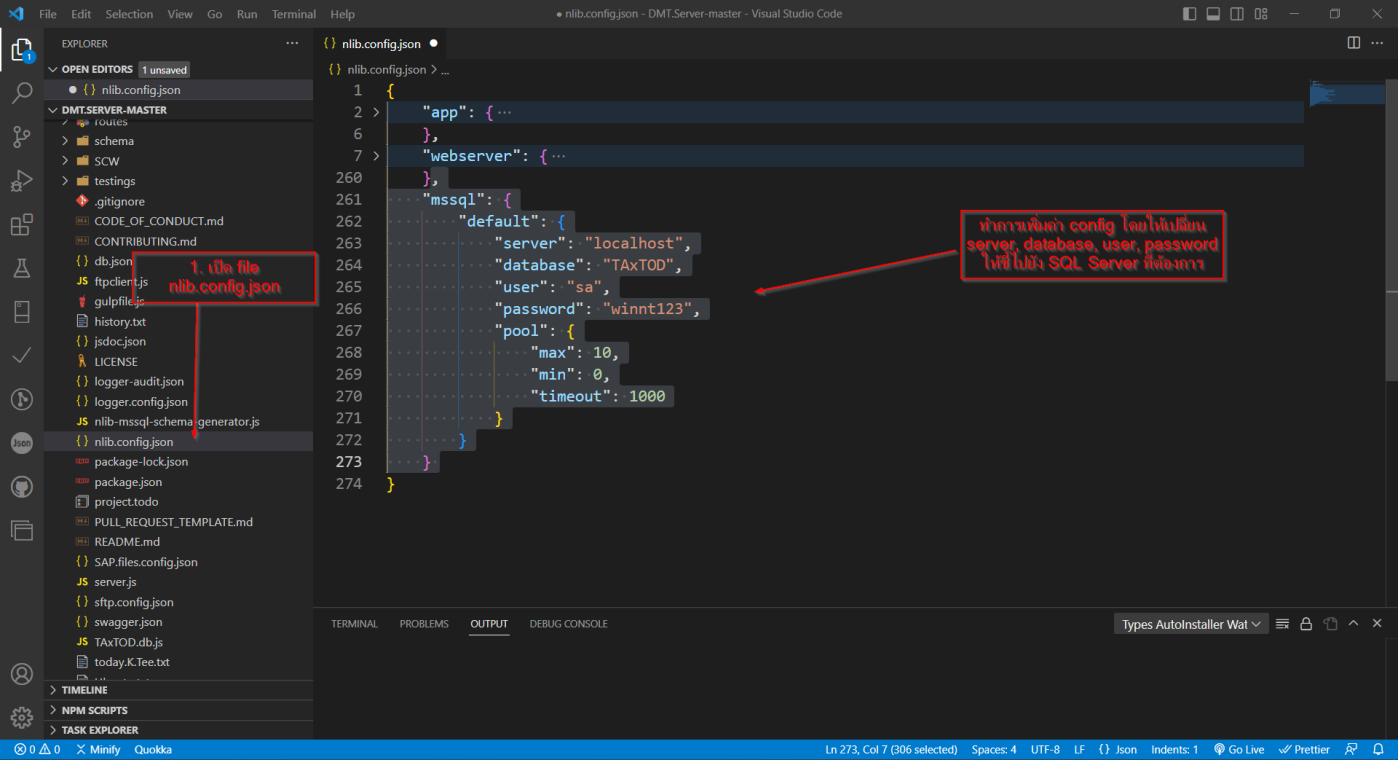
รอจน node.js ติดตั้ง package ต่าง ๆ ให้เสร็จ แล้วให้พิมพ์คำสั้ง node server.js ดังรูป



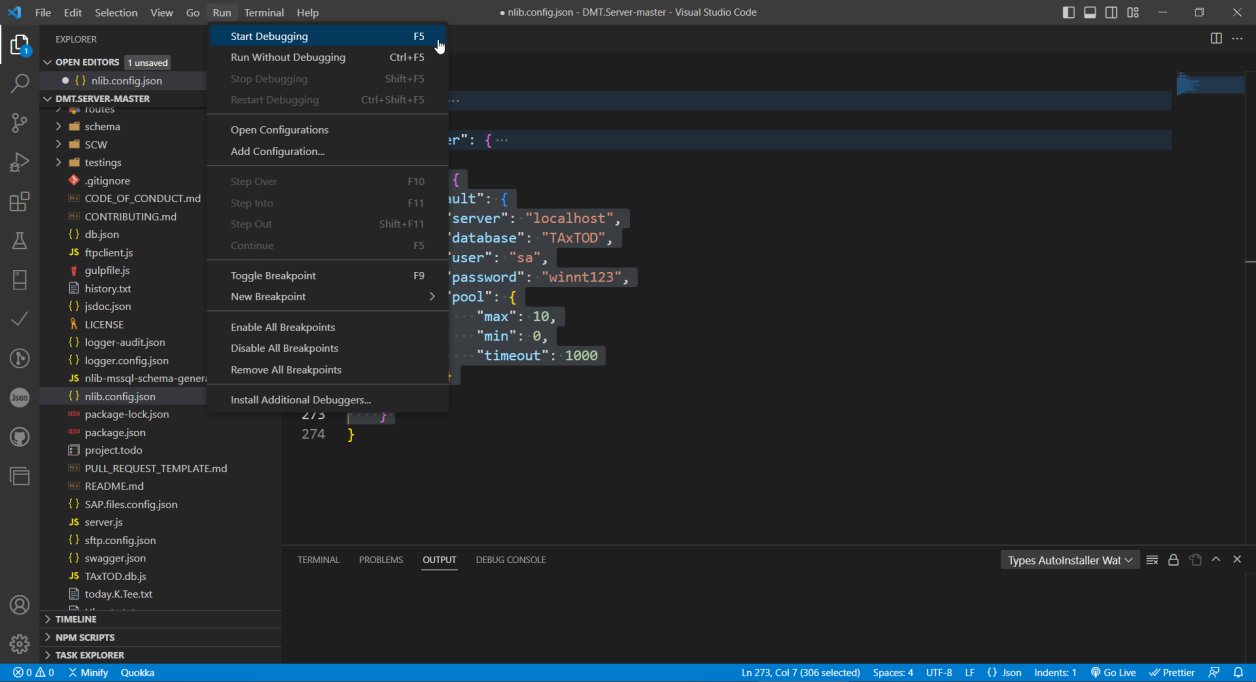
จากนั้นรอให้โปรแกรมทำงานจะแสดงหน้าจอดังนี้



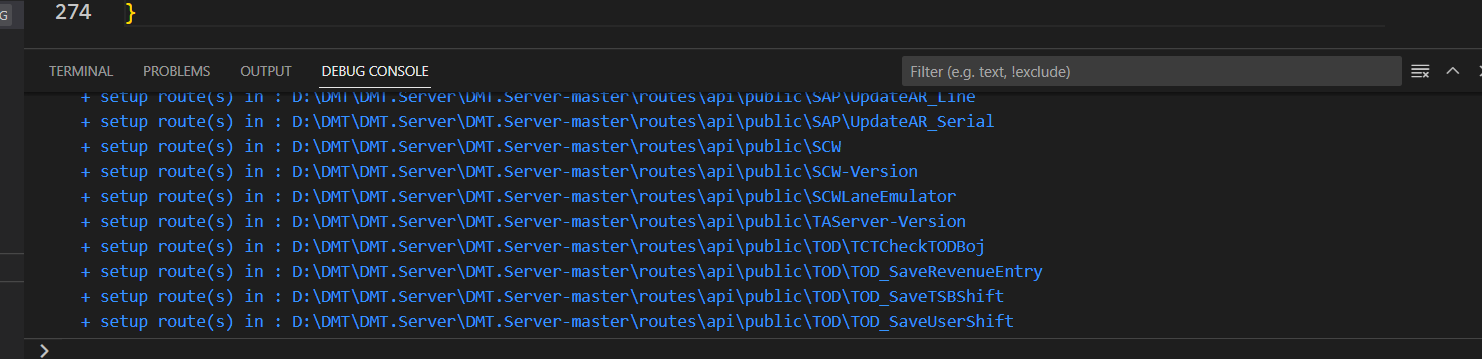
เมื่อถึงขั้นตอนนี้โปรแกรมสามารถทำงานได้แล้ว แต่ ณ. ตอนนี้ยังไม่ได้มีการตั้งค่าการเชื่อมต่อฐานข้อมูล ซึ่งต้องทำการหยุดการทำงานของโปรแกรมก่อน โดยกด Ctrl-C เพื่อหยุดการทำงาน จากนั้นให้กลับไปที่ Visual Studio code แล้วหา file ชื่อ nlib.config.json ดังรูป



เมื่อแก้ไขเสร็จแล้ว ให้ทำการ run ใหม่อีกครั้ง โดยใช้คำสั่ง node server.js ใน command prompt หรือ ทำการ run ผ่าน VS Code โดย เลือกจาก menu Run->Start Debugging หรือกด F5 ดังรูป



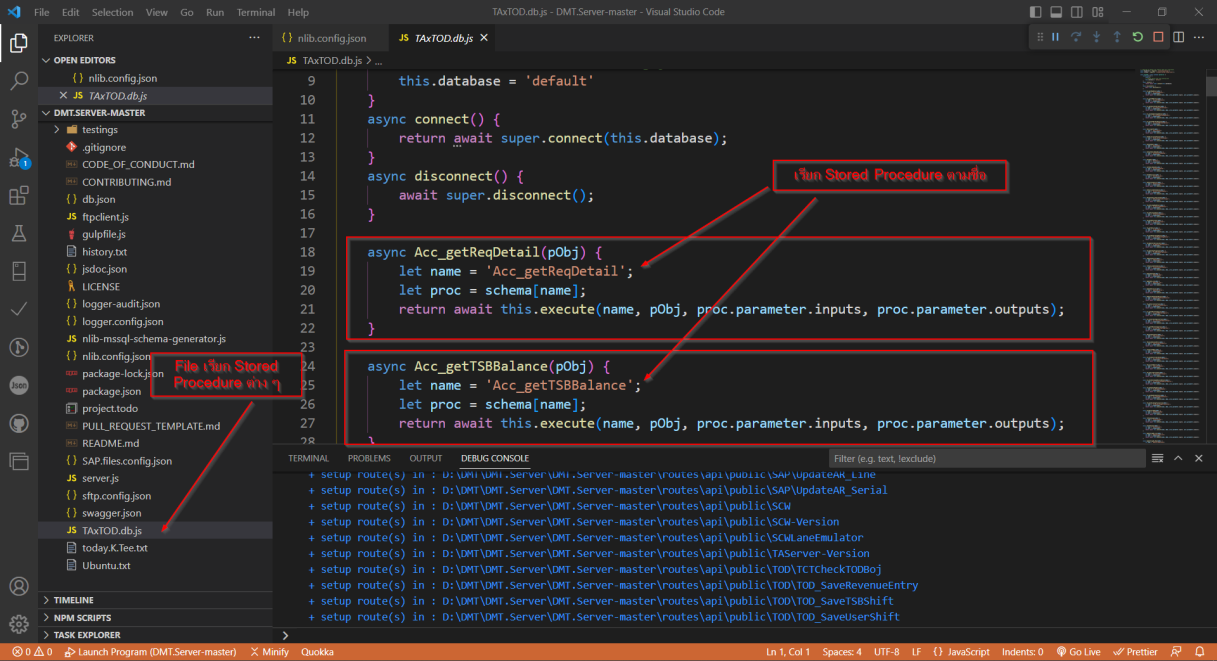
ซึ่งเมื่อสั่ง Run ไปแล้ว VS Code จะแสดง output ใน Tab Debug Console ด้านล่าง ดังรูป



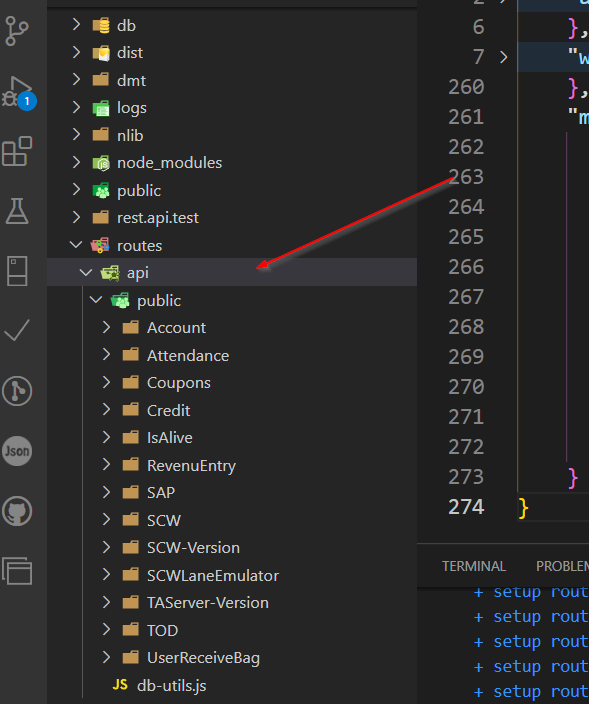
Function การทำงาน

TA Server จะมี function การทำงานหลัก ๆ อยู่ 2 ส่วนคือ

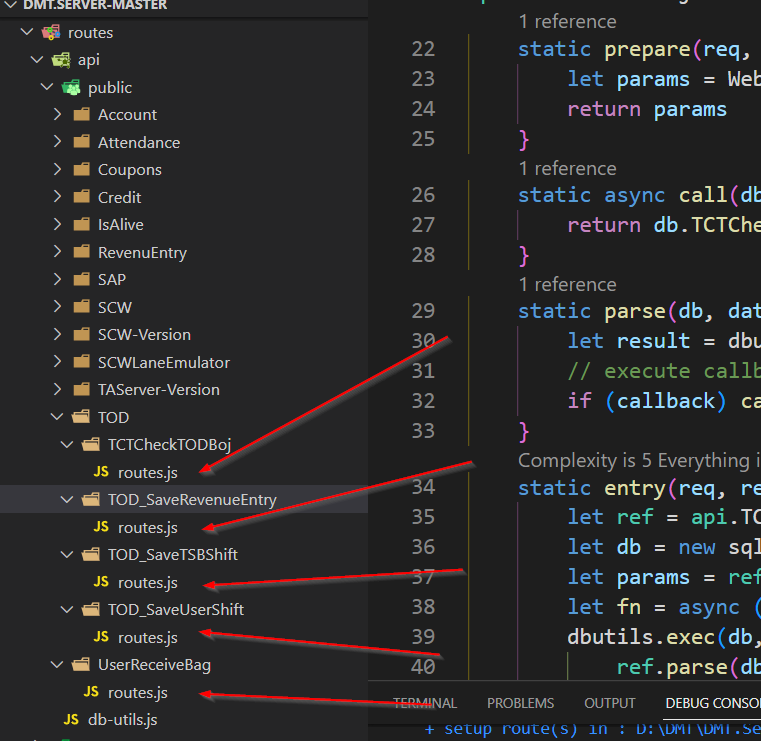
1. ส่วนการต่อกับระบบฐานข้อมูล เพื่ออ่านข้อมูล และบันทึกข้อมูล โดยในส่วนนี้ file ที่ใช้ทำงานหลักคือ file ชื่อ TAxTOD.db.js ซึ่งแสดงดังรูป



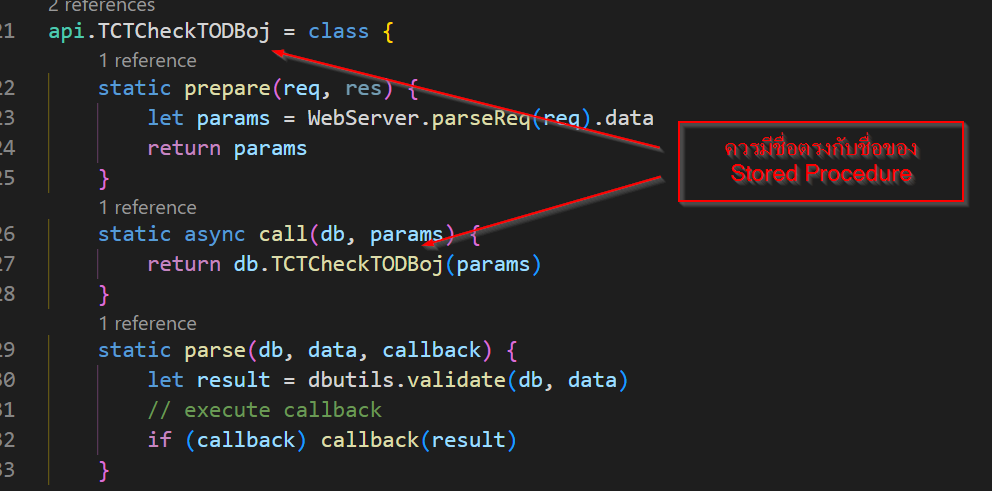
1. ส่วน REST API สำหรับให้ โปรแกรมภายนอกเรียกใช้ เช่น TOD, TA (Plaza), TA (Account) โดยในส่วนนี้ code การทำงานหลักจะอยู่ใน folder routes->api ดังรูป

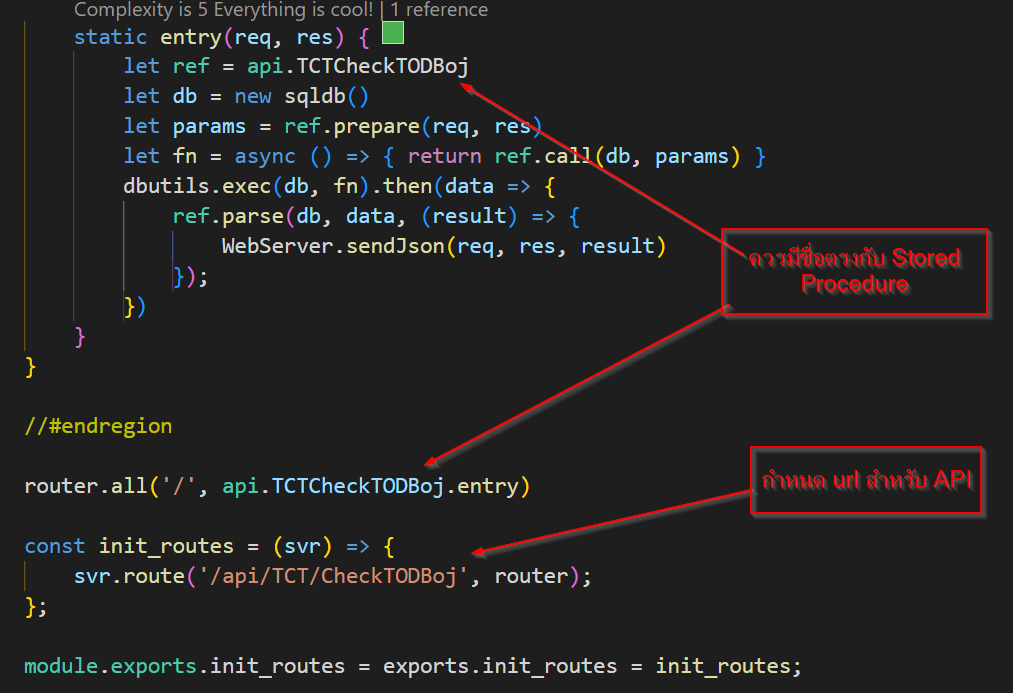


โดยในแต่ละ API จะมี file routes.js สำหรับการรองรับการเรียกใช้ โดย แสดงดังรูป โดยให้ดูจากลูกศรสีแดงจะเห็นว่าแต่ละ API จะมี route แยกกัน



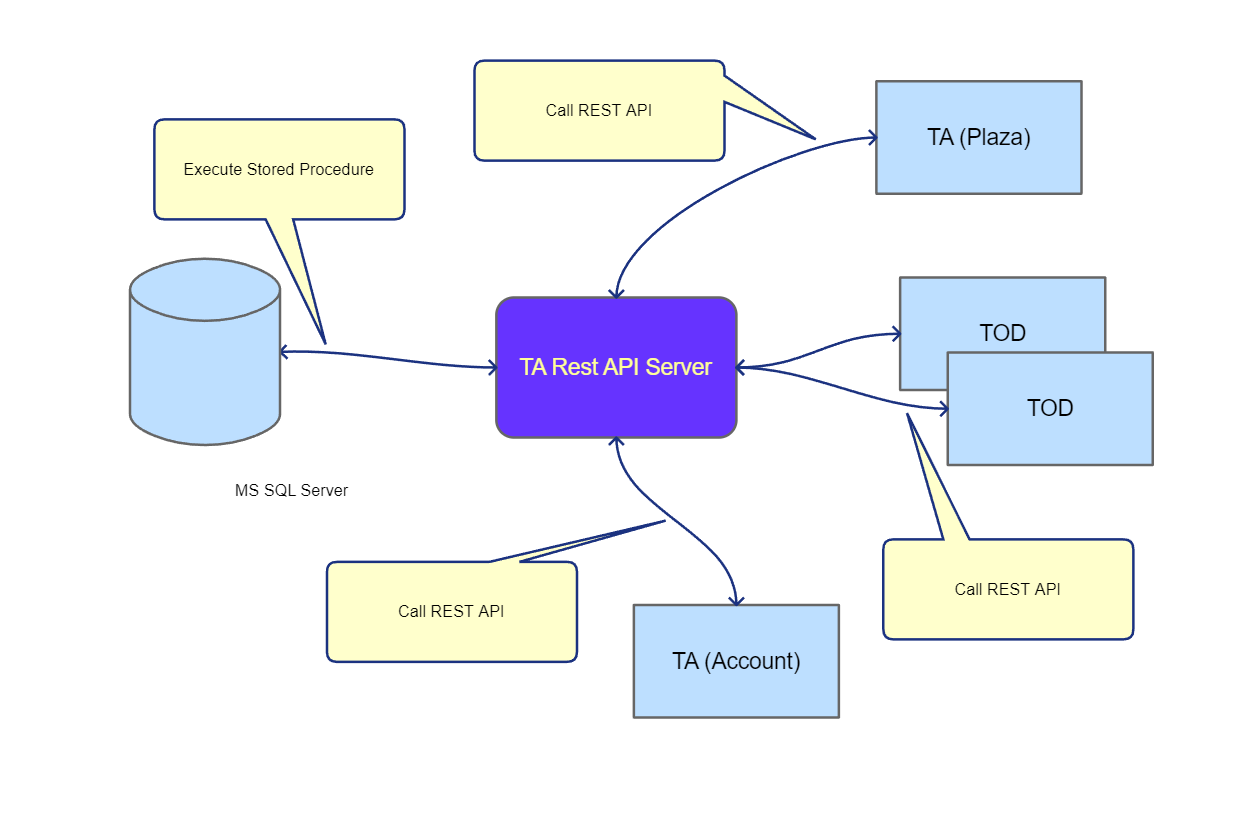
โดยแต่ละ route จะมี code ในการติดต่อกับ database ดังรูป





**Data Flow ของข้อมูล**

การทำงานของ TA Server แสดงได้ดังนี้



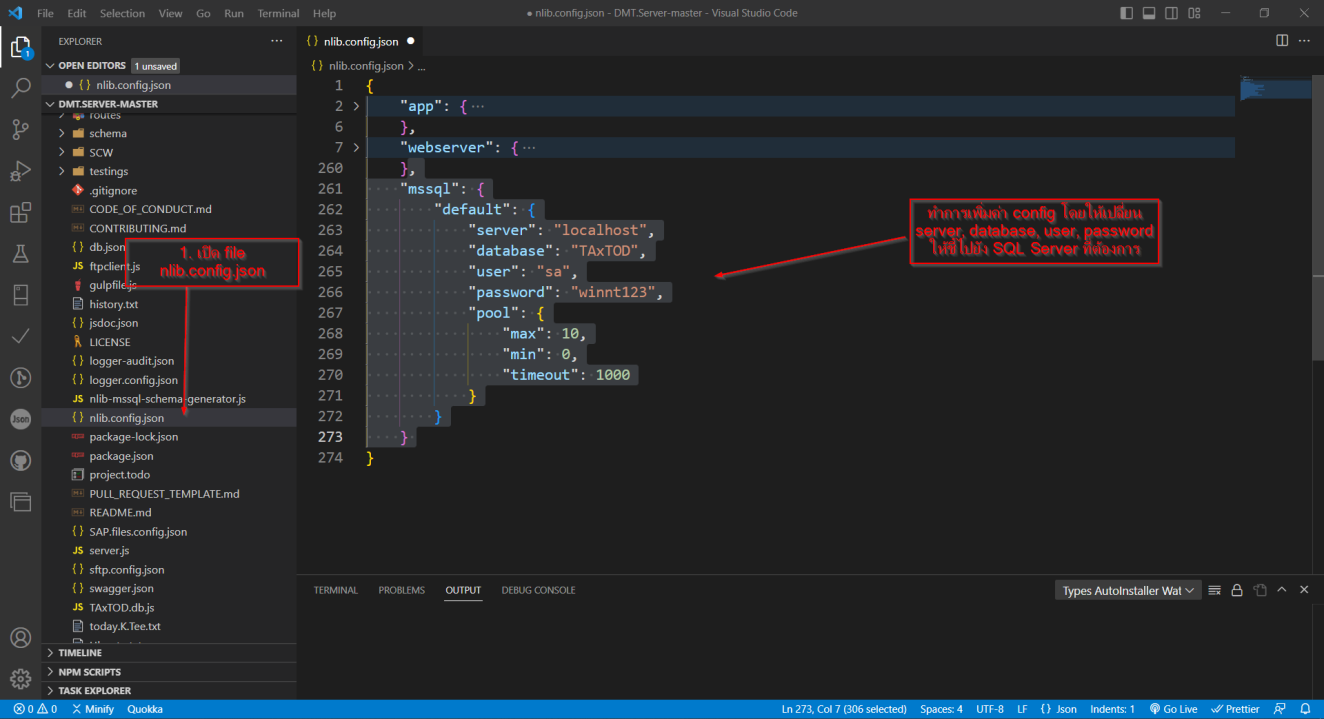
**การเชื่อมต่อ API และ Protocol**

ในการเชื่อมต่อกับระบบที่เกี่ยวข้องจะทำการเรียกผ่าน REST API โดย REST ย่อมาจาก Representational State Transfer เป็นรูปแบบการส่งข้อมูลระหว่าง Server-Client รูปแบบหนึ่งซ่ึ่งอยู่บนพื้นฐานของ HTTP Protocol เป็นการสร้าง Web Service เพื่อแลกเปลี่ยนข้อมูลกันผ่าน Application วิธีหนึ่ง ซึ่งส่งข้อมูลได้หลายชนิด ไม่ว่าจะเป็น Text, XML, JSON โดย ระบบที่ใช้ REST API ที่โปรแกรม TA Server มีให้เรียกใช้ จะประกอบด้วย

|  |  |  |  |
| --- | --- | --- | --- |
| Name | Save TSB Credit Balance | Method | POST |
| Url | http://localhost:8000/api/account/tsbcredit/save | | |
| Input |  | | |
| Output |  | | |
| Name | Gets TSB Credit Balance | Method | POST |
| Url | http://localhost:8000/api/account/tsbcredit | | |
| Input |  | | |
| Output |  | | |
| Name | Save User Credit on Job | Method | POST |
| Url | http://localhost:8000/api/account/UserCredit/save | | |
| Input |  | | |
| Output |  | | |
| Name | Gets User On Job Credit by TSB | Method | POST |
| Url | http://localhost:8000/api/account/usercredit/search | | |
| Input |  | | |
| Output |  | | |
| Name | Save Request Head | Method | POST |
| Url | http://localhost:8000/api/account/request/save | | |
| Input |  | | |
| Output |  | | |
| Name | Save Request Detail | Method | POST |
| Url | http://localhost:8000/api/account/request/savedetail | | |
| Input |  | | |
| Output |  | | |
| Name | Account get request list | Method | POST |
| Url | http://localhost:8000/api/account/request/getlist | | |
| Input |  | | |
| Output |  | | |
| Name | Account get request detail | Method | POST |
| Url | http://localhost:8000/api/account/request/getdetail | | |
| Input |  | | |
| Output |  | | |
| Name | Account Save Approve request detail | Method | POST |
| Url | http://localhost:8000/api/account/request/approve | | |
| Input |  | | |
| Output |  | | |
| Name | TSB get Approve request update | Method | POST |
| Url | http://localhost:8000/api/account/request/getapprove | | |
| Input |  | | |
| Output |  | | |
| Name | TSB get Approve request update | Method | POST |
| Url | http://localhost:8000/api/account/request/getapprovedetail | | |
| Input |  | | |
| Output |  | | |
| Name | TSB get Approve request detail | Method | POST |
| Url | http://localhost:8000/api/account/request/getapprovedetail | | |
| Input |  | | |
| Output |  | | |
| Name | Get Approve TSB Credit List | Method | POST |
| Url | http://localhost:8000/api/account/appcredit/list | | |
| Input |  | | |
| Output |  | | |
| Name | Get TSB Aprove Credit Transaction List | Method | POST |
| Url | http://localhost:8000/api/account/appcredit/translist | | |
| Input |  | | |
| Output |  | | |
| Name | Save TSB Aprove Credit Transaction | Method | POST |
| Url | http://localhost:8000/api/account/appcredit/savetrans | | |
| Input |  | | |
| Output |  | | |
| Name | Account Save Received request detail | Method | POST |
| Url | http://localhost:8000/api/account/request/receive | | |
| Input |  | | |
| Output |  | | |
| Name | Gets all user coupons that assigned by TA | Method | POST |
| Url | http://localhost:8000/api/users/coupons/search | | |
| Input |  | | |
| Output |  | | |
| Name | Send user coupon sold to TA server | Method | POST |
| Url | http://localhost:8000/api/TCT/coupons/sold | | |
| Input |  | | |
| Output |  | | |
| Name | Gets TSB list | Method | POST |
| Url | http://localhost:8000/api/account/sap/tsblist | | |
| Input |  | | |
| Output |  | | |
| Name | Gets Customer Code list | Method | POST |
| Url | http://localhost:8000/api/account/sap/getcustomer | | |
| Input |  | | |
| Output |  | | |
| Url | http://localhost:8000/api/account/sap/couponsoldlist | | |
| Input |  | | |
| Output |  | | |

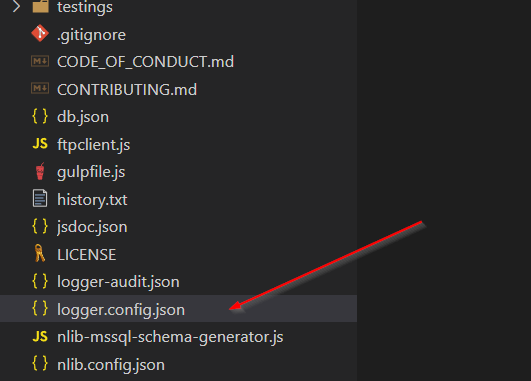
**การเชื่อมต่อ Database**

ในการเชื่อมต่อ database ให้ทำการหา file nlib.config.json ตามรูป และทำการแก้ไข config เพื่อชี้ไปยังตำแหน่งที่ Database Server ที่ต้องการ

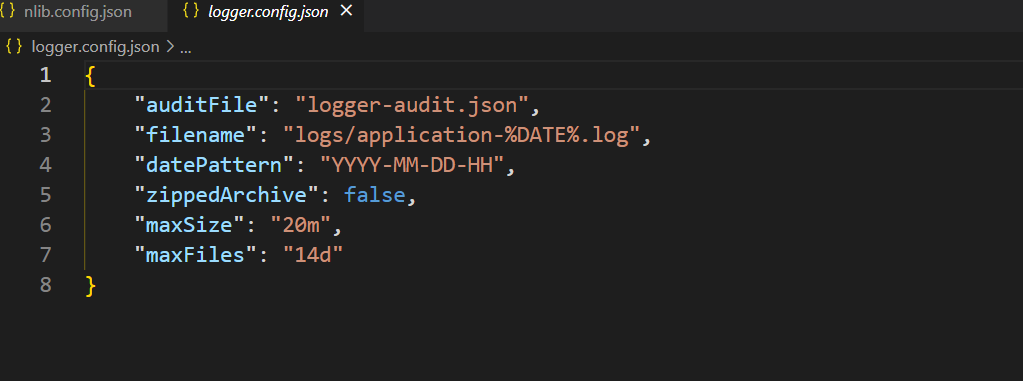


**การ Log**

การตั้งค่า configuration ในการเก็บ log file สามารถ แก้ไขได้ที่ file logger.config.json ดังรูป



โดยข้อมูลใน file จะแสดงดังรูป ต่อไปนี้



**การติดตั้ง Software ที่ build บน production**

ในการติดตั้ง Software ที่ production site สามารถทำตามขั้นตอนในหัวข้อ **การติดตั้ง Editor, Compiler, Component** ได้ เนื่องจาก node.js จะไม่มีการ compile เป็น executable files จำเป็นต้องนำ code ทั้ง folderไป copy ทับและ ติดตั้ง package ต่าง ๆ ตามขั้นตอนในหัวข้อดังกล่าว และทำการเรียกทำงานโดยทำผ่าน command prompt โดยใช้คำสั่ง node server.js ดังรูป

